CrimeBoard (Use Cases, SQL Query & Relational Algebra Expression)

Class: DAMG6210
CRN: 16149
Team members:
Sayeed Ahmed - 002191535
Natarajan Lekshmi Narayana Pillai - 002766033
Chaman Betrabet – 002784662
Sunil Kumar Rudrakumar - 002764807

1)

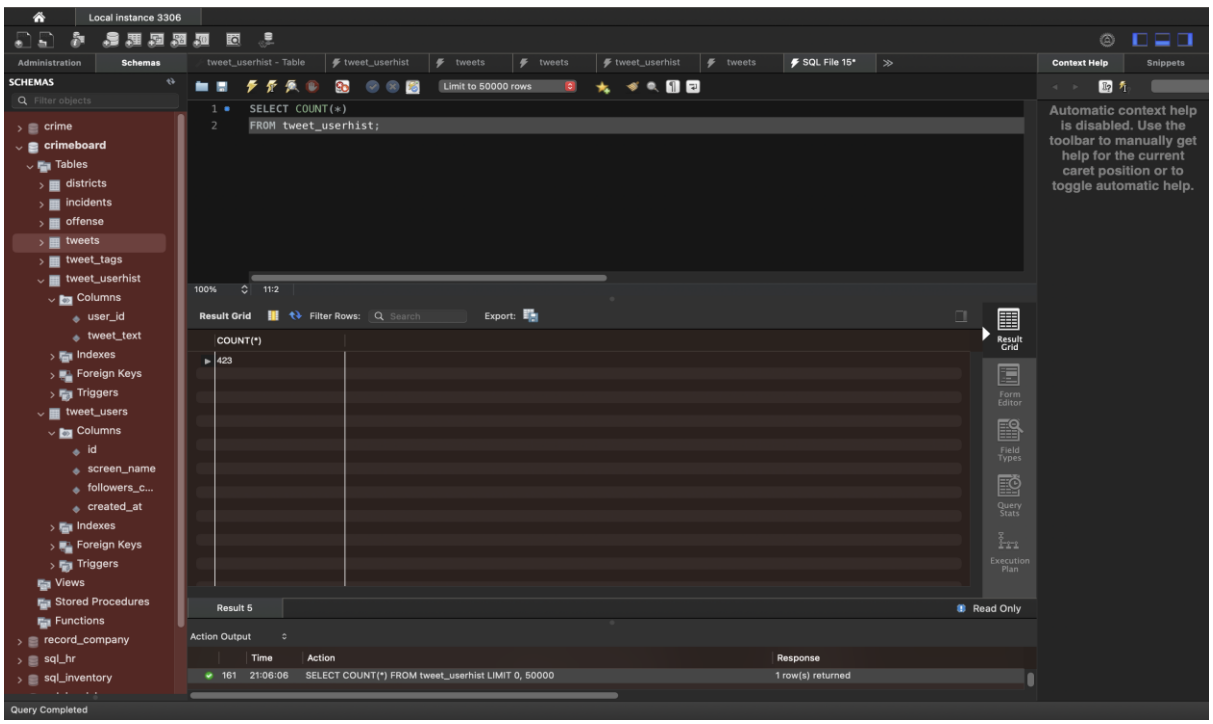
Use Case: How many tweets did the users make in the past 24 hours?
Description: Use the tweet_userhist table to show the number of tweets made in 24hrs by a user.
Actor: User
Actor action: The user views the number of tweets made in 24hrs by a user.
System Responses: the number of tweets made in 24hrs by a user is displayed
Post Condition: system displays the number of tweets made in 24hrs by a user.

SQL Query

SELECT COUNT(*) FROM tweet_userhist;

Relational Algebra Expression

$\pi_{tweet\_count}\ tweet\_userhist$



2)

Use Case: What's the favourite count of a tweet

Description: Use the tweets table to show the number of favourite count of a tweet.

Actor: User

Actor action: The user views the number of the favourite count of a tweet.

System Responses: The number of the favourite count of a tweet is displayed

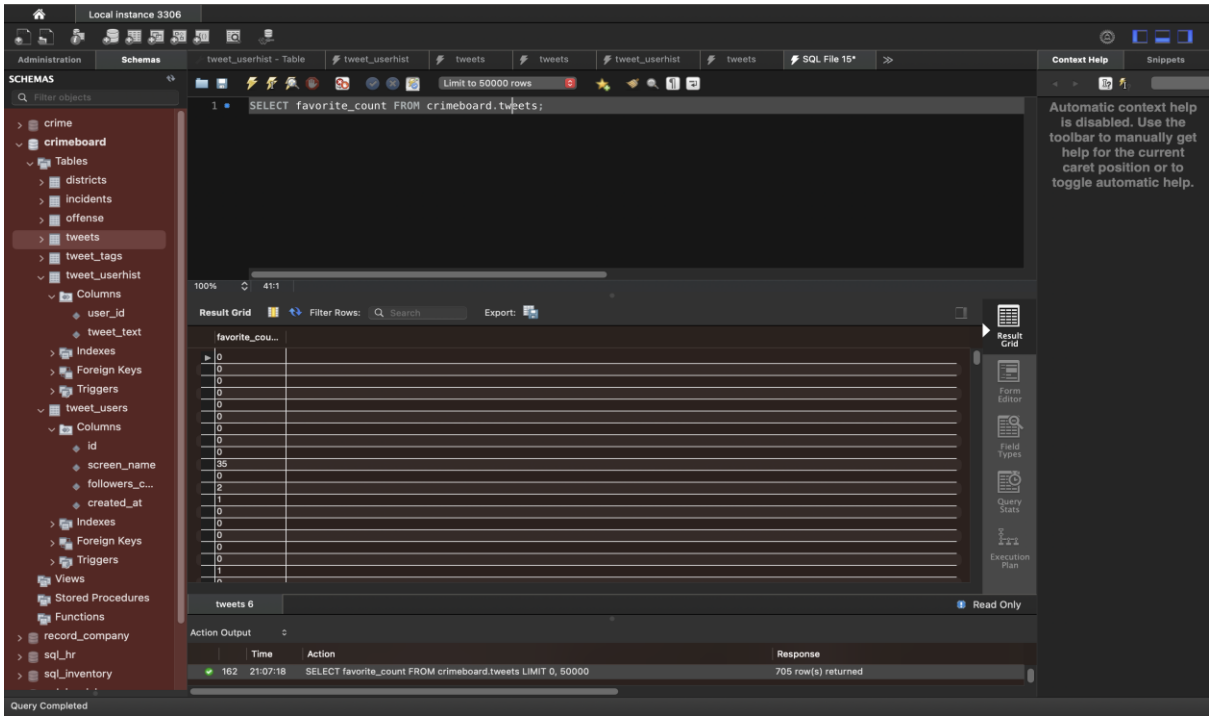
Post Condition: system displays the number of favourite count of a tweet.

SQL Query

SELECT favorite_count FROM tweets;

Relational Algebra Expression

$\pi_{favorite\_count}\ tweets$



3)

Use Case: What are the retweet counts of tweets?

Description: Use the tweets table to show the number of retweets of tweets.

Actor: User

Actor action: The user views the number of retweets of tweets.

System Responses: The number of retweets of tweets is displayed

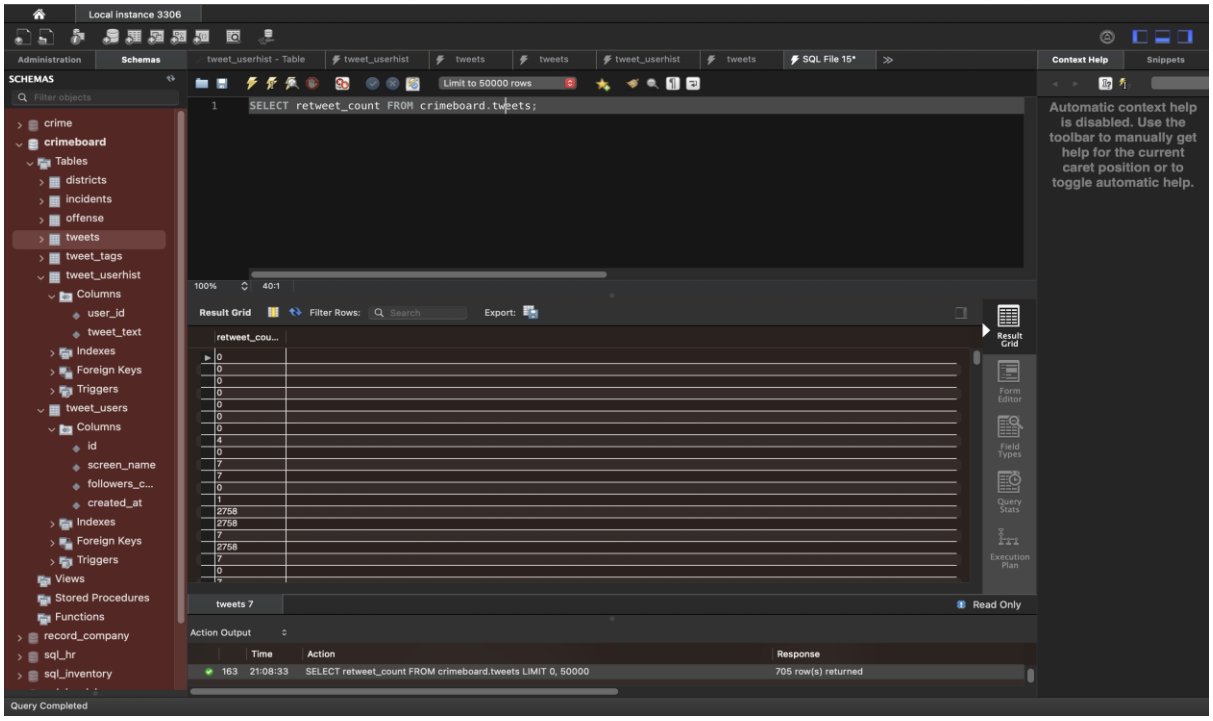
Post Condition: system displays the number of retweets of tweets.

SQL Query

SELECT retweet_count FROM tweets;

Relational Algebra Expression

$\pi_{retweet\_count}\ tweets$



4)

Use Case: When was the latest tweet created?

Description: Use the tweets table to show the latest tweet created.

Actor: User

Actor action: The user views the latest tweet created.

System Responses: The latest tweet created is displayed

Post Condition: system displays the latest tweet created.

SQL Query

SELECT id_str, text, id, max(created_at)

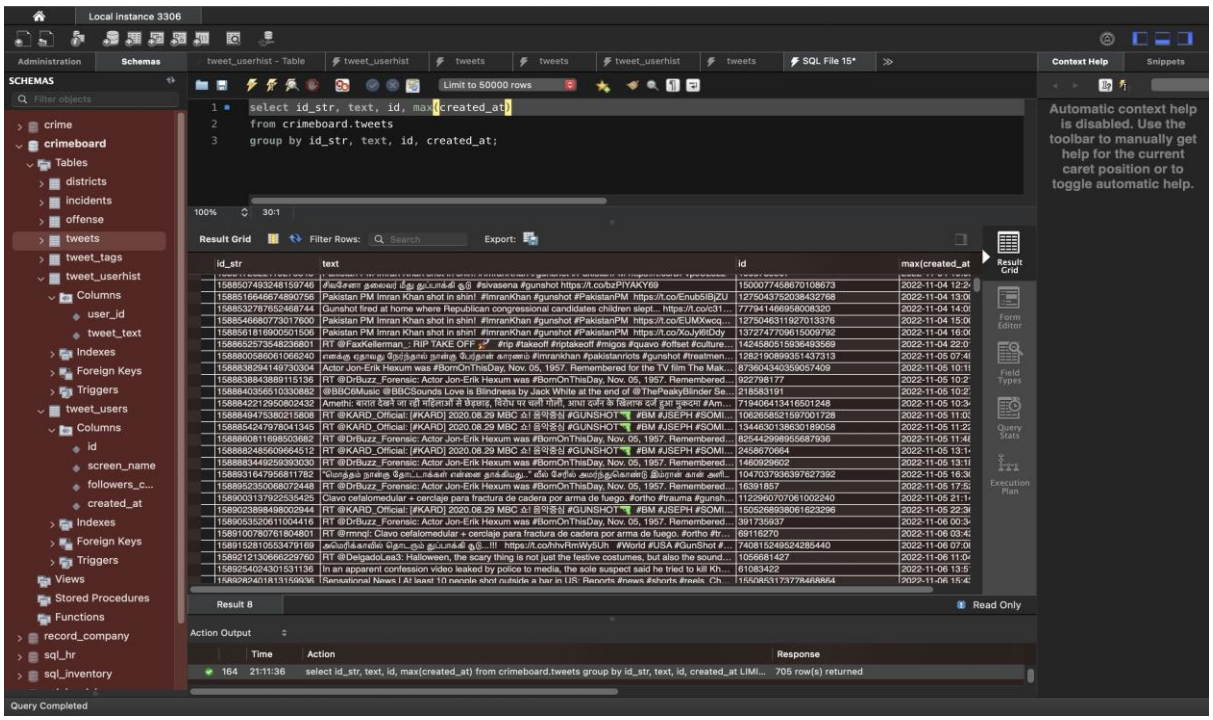
FROM tweets

GROUP BY id_str, text, id, created_at

Relational Algebra Expression

π *id_str, text, id, MAX (created_at)*

γ *id_str, text, id, created_at, MAX (created_at)* *tweets*



5)

Use Case: Does a single user tweet multiple times?

Description: Use the tweets table to show does a single user tweets multiple times.

Actor: User

Actor action: The user views a single user's tweet multiple times.

System Responses: does a single user tweet multiple times is displayed

Post Condition: system displays does a single user tweets multiple time

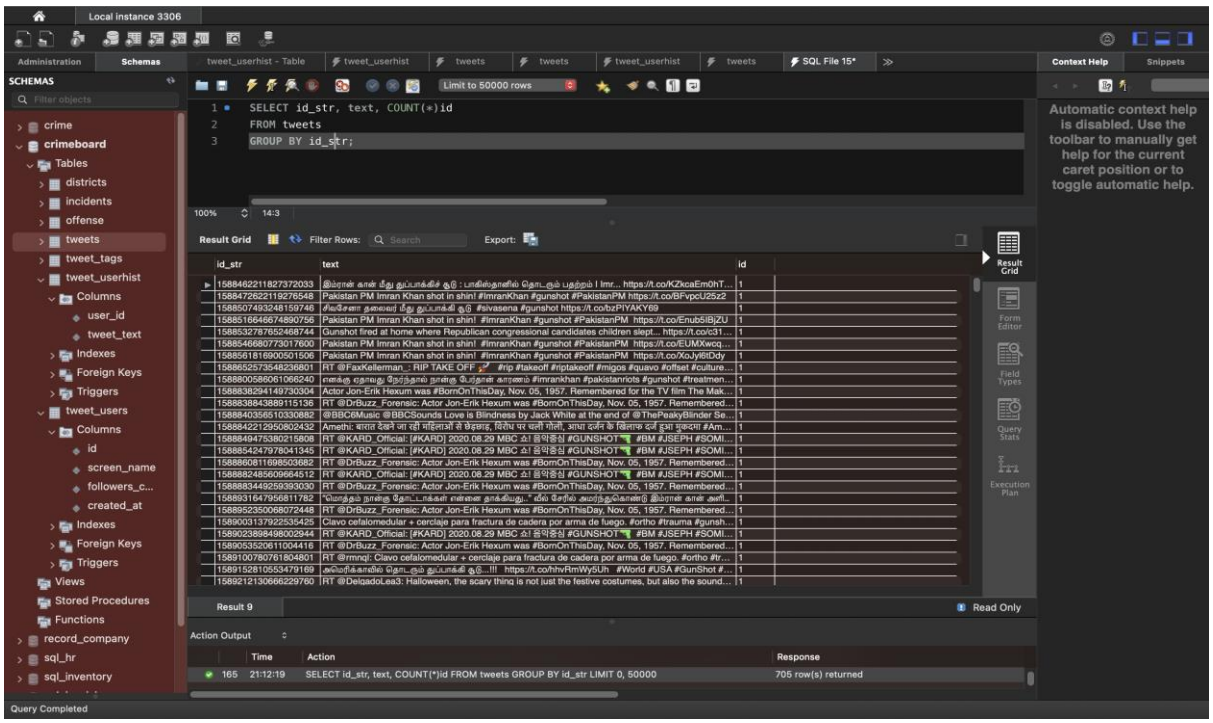
SQL Query

SELECT id_str, text, COUNT(*)id
FROM tweets
GROUP BY id_str

Relational Algebra Expression

π *id_str, text, id*

γ *id_str*, *tweets*



6)

Use Case: How many users tweeted about crime?

Description: Use the tweet_users table to show how many users tweeted about crime.

Actor: User

Actor action: The user views how many users tweeted about crime.

System Responses: How many users tweeted about the crime is displayed.

Post Condition: system displays How many users tweeted about crime.

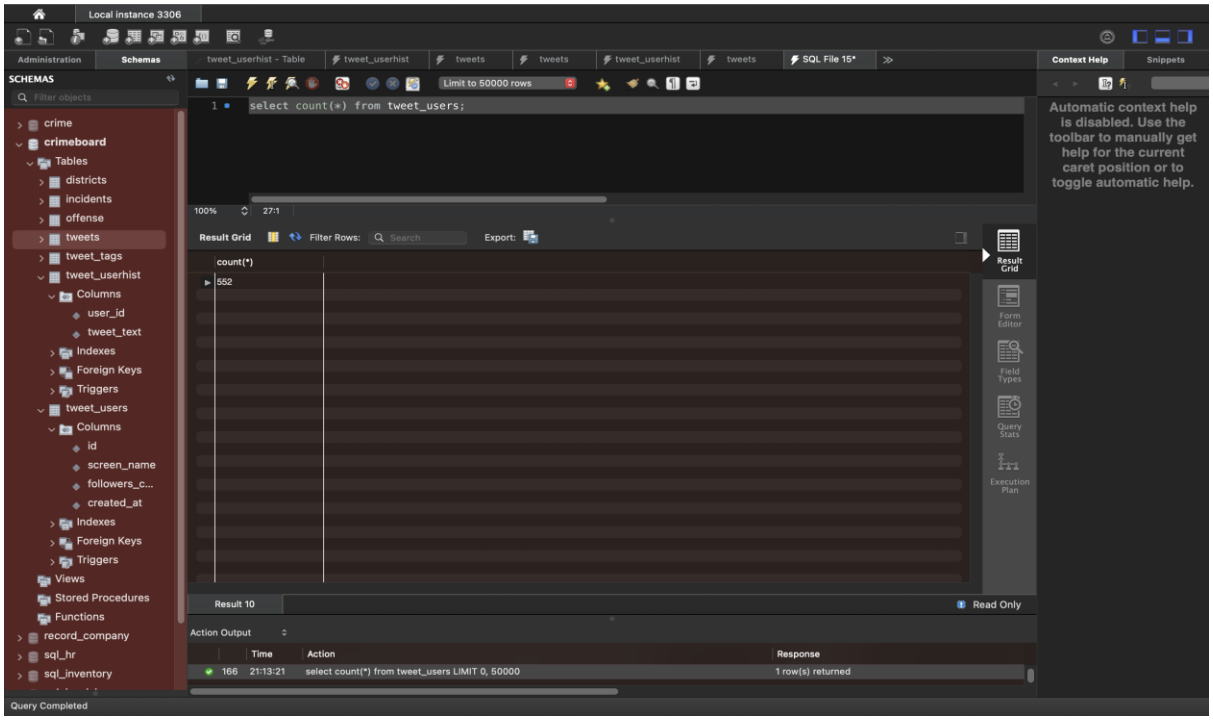
SQL Query

SELECT COUNT(*) FROM tweet_users

Relational Algebra Expression

π *COUNT (*)*

γ *COUNT (*)* *tweet_users*



7)

Use Case: How many tweets have the keyword crime?

Description: Use the tweets table to show how many tweets have the keyword crime.

Actor: User

Actor action: The user views how many tweets have the keyword crime

System Responses: How many tweets have the keyword crime are displayed.

Post Condition: system displays how many tweets that have the keyword crime

SQL Query

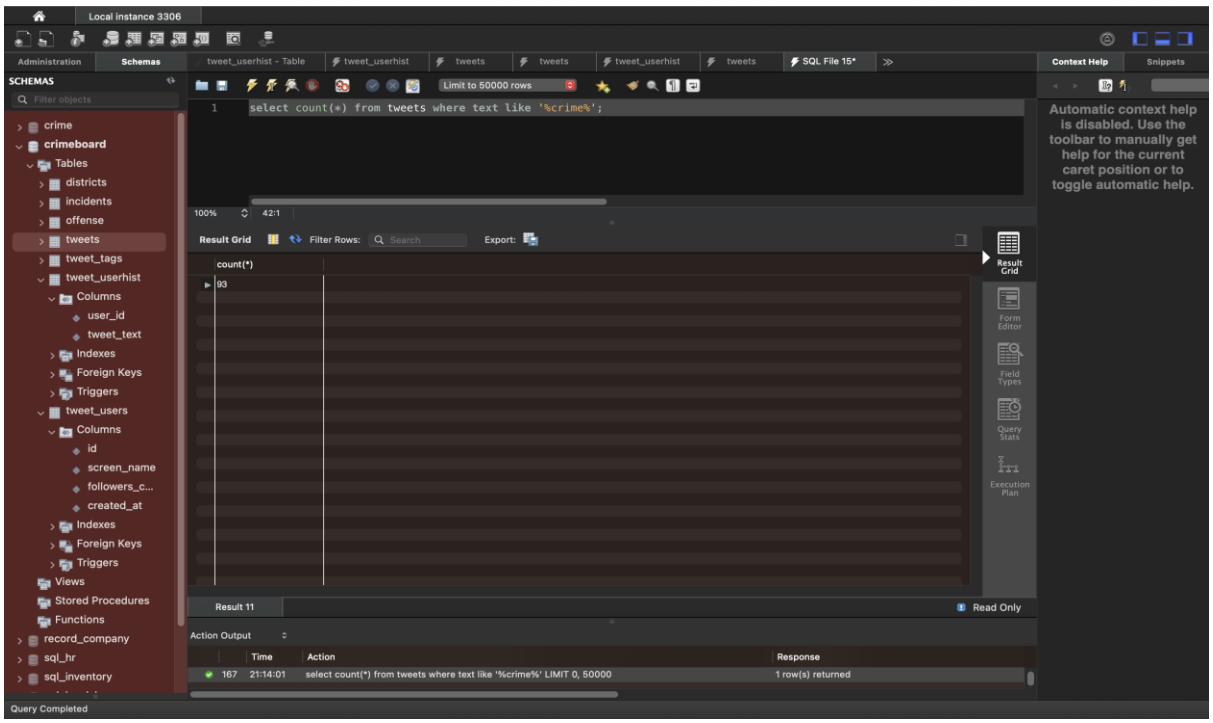
SELECT COUNT(*) FROM tweets WHERE TEXT LIKE '%crime%'

Relational Algebra Expression

π *COUNT (*)*

γ *COUNT (*)*

σ *text LIKE "%crime%"* *tweets*



8)

Use Case: Is there a police department official handle tweeting in our records?

Description: Use the tweet_users table to show whether Is there a police department official handling tweeting in our records.

Actor: User

Actor action: The user views whether there is a police department official handle tweeting in our records.

System Responses: Is there a police department official handle tweeting in our records displayed.

Post Condition: system displays whether there is a police department official handle tweeting in our records

SQL Query

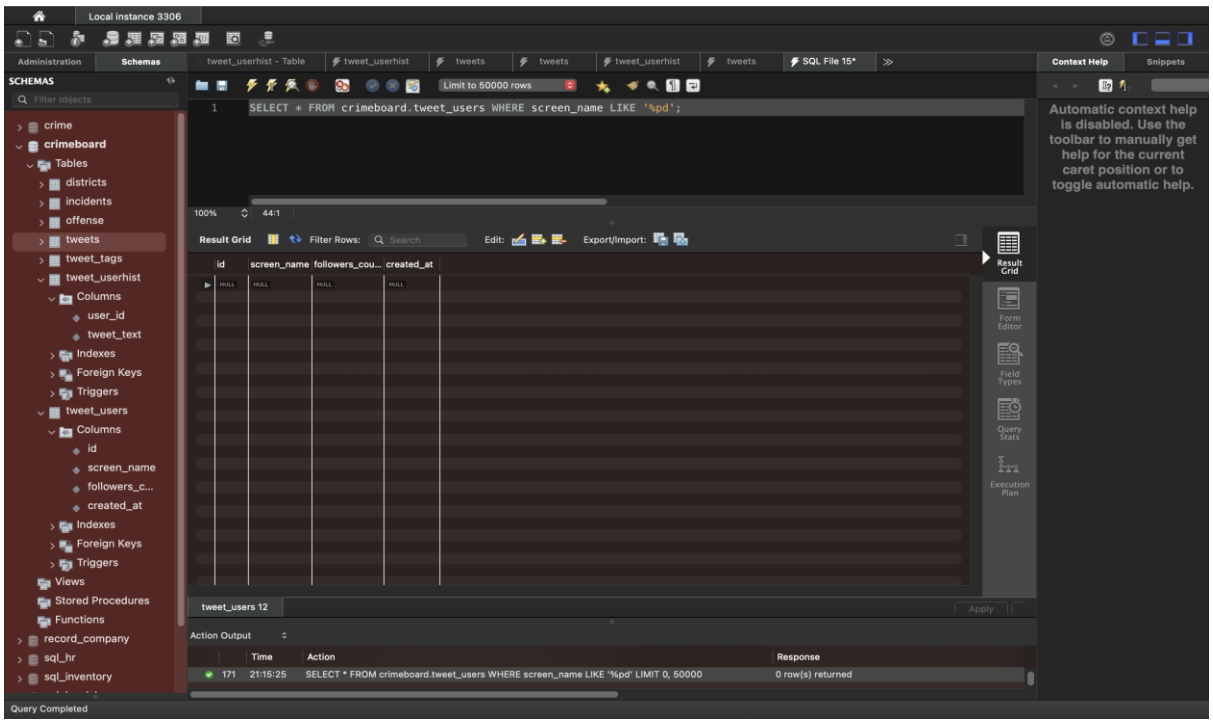
*SELECT * FROM tweet_users WHERE screen_name LIKE '%pd'*

Relational Algebra Expression

π *COUNT (*)*

γ *COUNT (*)*

σ *screen_name LIKE "%PD%"* *tweet_users*



9)

Use Case: Which city had the most tweets on crime?

Description: Use the tweets table to show which city had the most tweets on crime.

Actor: User

Actor action: The user views which city had the most tweets on crime

System Responses: which city had the most tweets on crime are displayed.

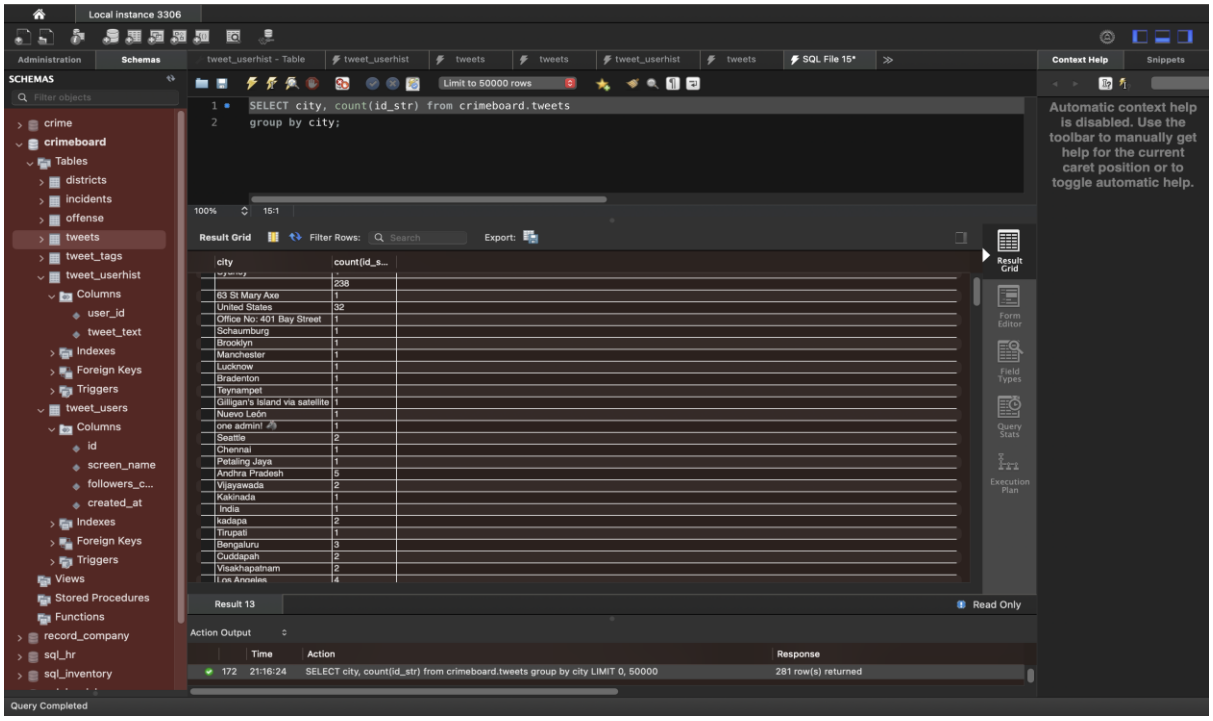
Post Condition: system displays which city had the most tweets on crime

SQL Query

SELECT city, count(id_str) FROM tweets
GROUP BY city;

Relational Algebra Expression

$\gamma_{city,\ COUNT\ (id\_str)}$ *tweets*



10)

Use Case: How many tweets were tweeted on the current date?

Description: Use the tweets table to show how many tweets were tweeted on the current date

Actor: User

Actor action: The user views how many tweets were tweeted on the current date.

System Responses: how many tweets were tweeted on the current date is displayed.

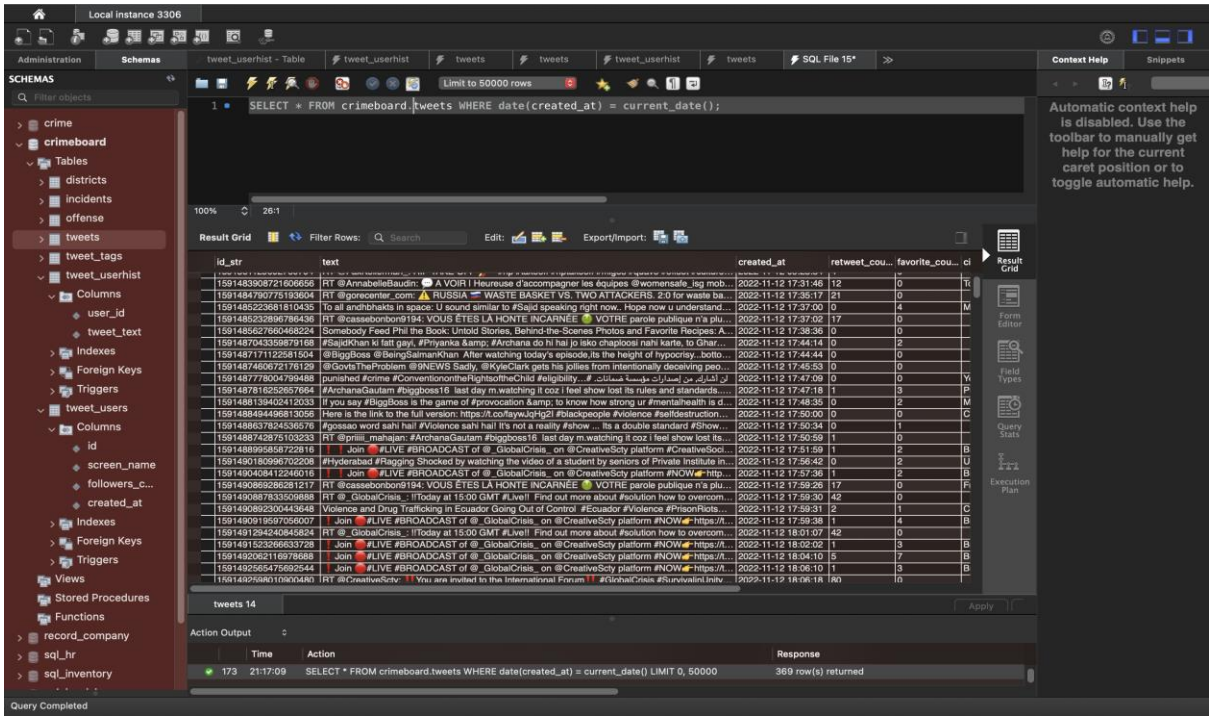
Post Condition: system displays how many tweets were tweeted on the current date.

SQL Query

*SELECT * FROM tweets WHERE date(created_at) = current_date();*

Relational Algebra Expression

$\sigma_{created\_at = current\_date}\ tweets$



11)

Use Case: How many tweets were tweeted at the current hour?

Description: Use the tweets table to show how many tweets were tweeted on the current date

Actor: User

Actor action: The user views how many tweets were tweeted on the current date.

System Responses: how many tweets were tweeted on the current date is displayed.

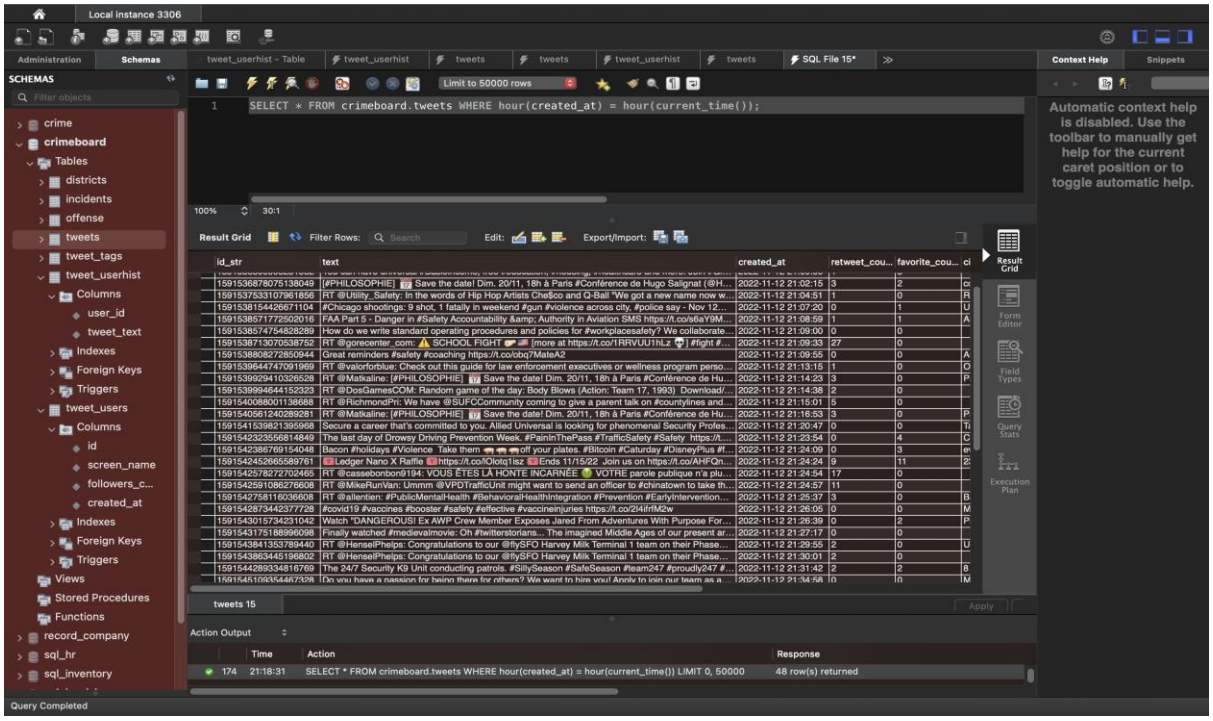
Post Condition: system displays how many tweets were tweeted on the current date.

SQL Query

*SELECT * FROM tweets WHERE hour(created_at) = hour(current_time());*

Relational Algebra Expression

$\sigma_{created\_at = hour}\ tweets$



12)

Description: What date had the most tweets about crime?

Actor: User

Actor action: The user views what date had the most tweets about crime

System Responses: what date had the most tweets about crime
are displayed.

Post Condition: system displays what date had the most tweets about crime

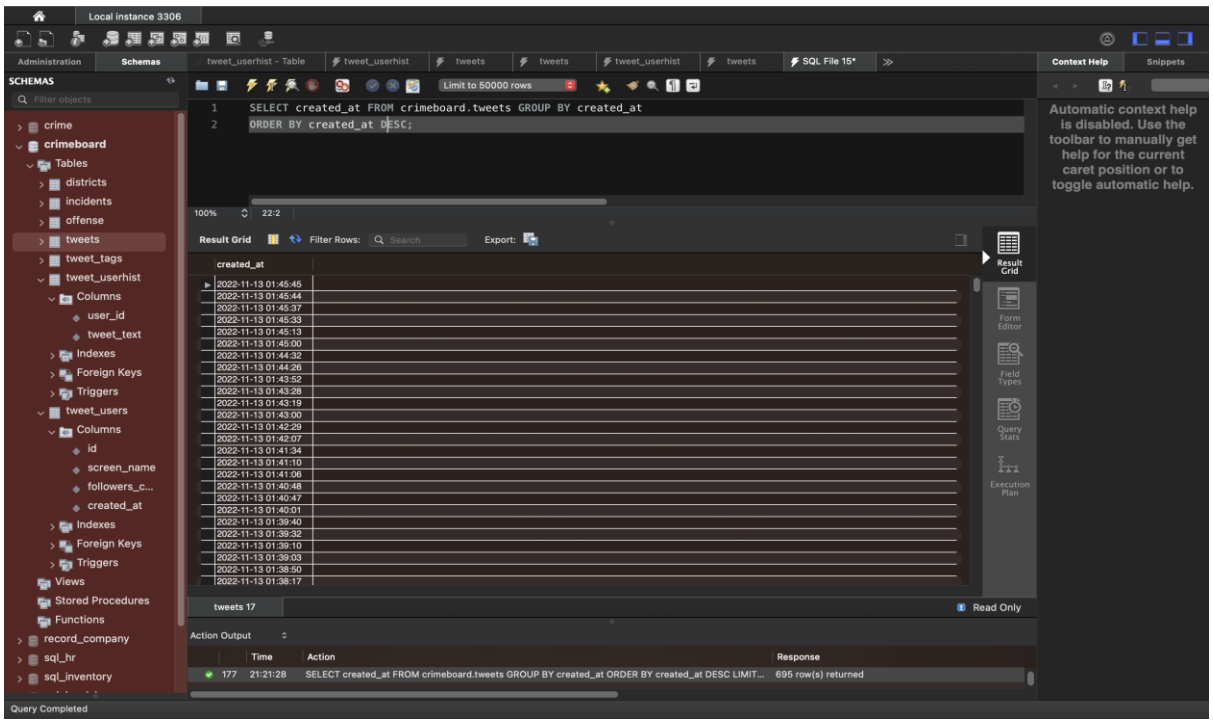
SQL Query

SELECT created_at FROM tweets GROUP BY created_at
ORDER BY created_at DESC;

Relational Algebra Expression

τ *created_at* $\downarrow$

γ *created_at, COUNT (created_at)* *tweets*



13)

Use Case: Which city had the most tweets on crime?

Description: Use the tweets table to show which city had the most tweets on crime.

Actor: User

Actor action: The user views which city had the most tweets on crime.

System Responses: which city had the most tweets on crime is displayed.

Post Condition: system displays which city had the most tweets on crime.

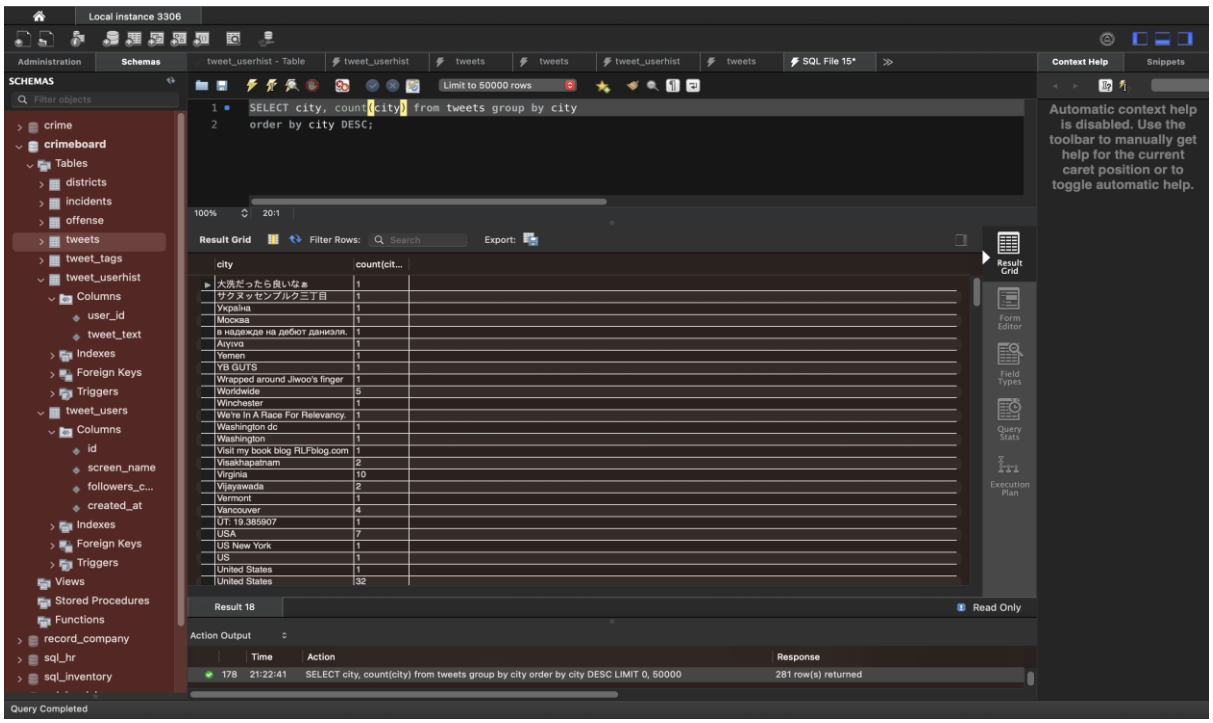
SQL Query

SELECT city, count(city) from tweets group by city
order by city DESC;

Relational Algebra Expression

$\tau_{city \downarrow}$

$\gamma_{city, COUNT(city)}$ *tweets*



14)

Use Case: Which state had the most tweets on crime?

Description: Use the tweets table to show which state had the most tweets on crime.

Actor: User

Actor action: The user views which state had the most tweets on crime.

System Responses: which state had the most tweets on crime is displayed.

Post Condition: system displays which state had the most tweets on crime.

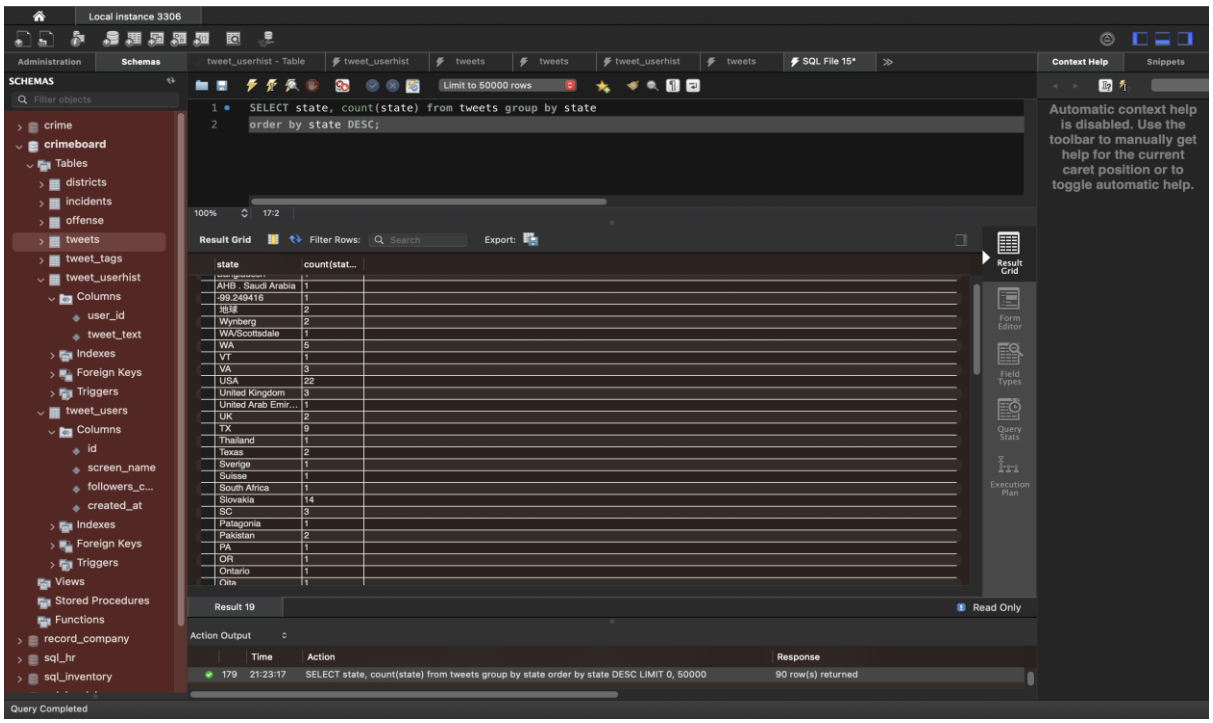
SQL Query

SELECT state, count(state) from tweets group by state
order by state DESC;

Relational Algebra Expression

τ *state* ↓

γ *state, COUNT (state)* *tweets*



15)

Use Case: Which country had the most tweets on crime?

Description: Use the tweets table to show which country had the most tweets on crime.

Actor: User

Actor action: The user views which country had the most tweets on crime.

System Responses: which country had the most tweets on crime is displayed.

Post Condition: system displays which country had the most tweets on crime.

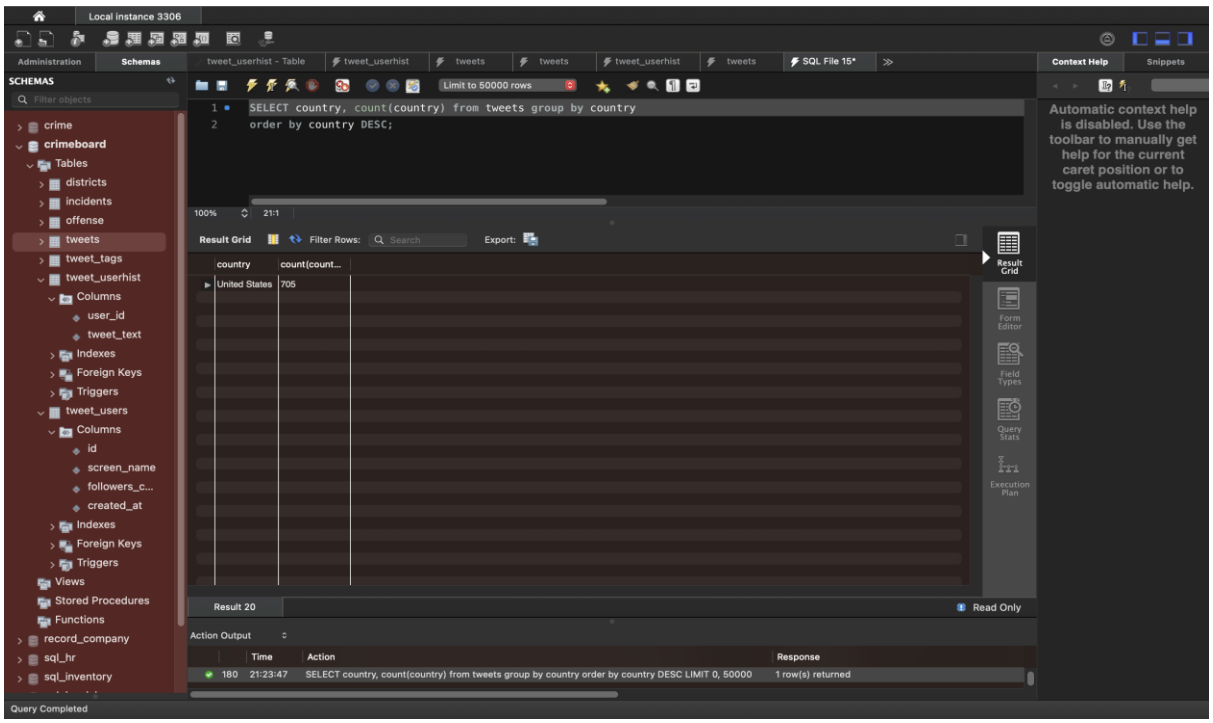
SQL Query

SELECT country, count(country) from tweets group by country
order by country DESC;

Relational Algebra Expression

τ *country* $\downarrow$

γ *country, COUNT (country)* *tweets*



16)

Use Case: How many tweets about crime were sent between the hours of 6 pm and 6 am?

Description: Use the tweets table to show how many tweets about crime were sent between the hours of 6 pm and 6 am.

Actor: User

Actor action: The user views how many tweets about crime were sent between the hours of 6 pm and 6 am.

System Responses: how many tweets about crime were sent between the hours 6 pm and 6 am is displayed.

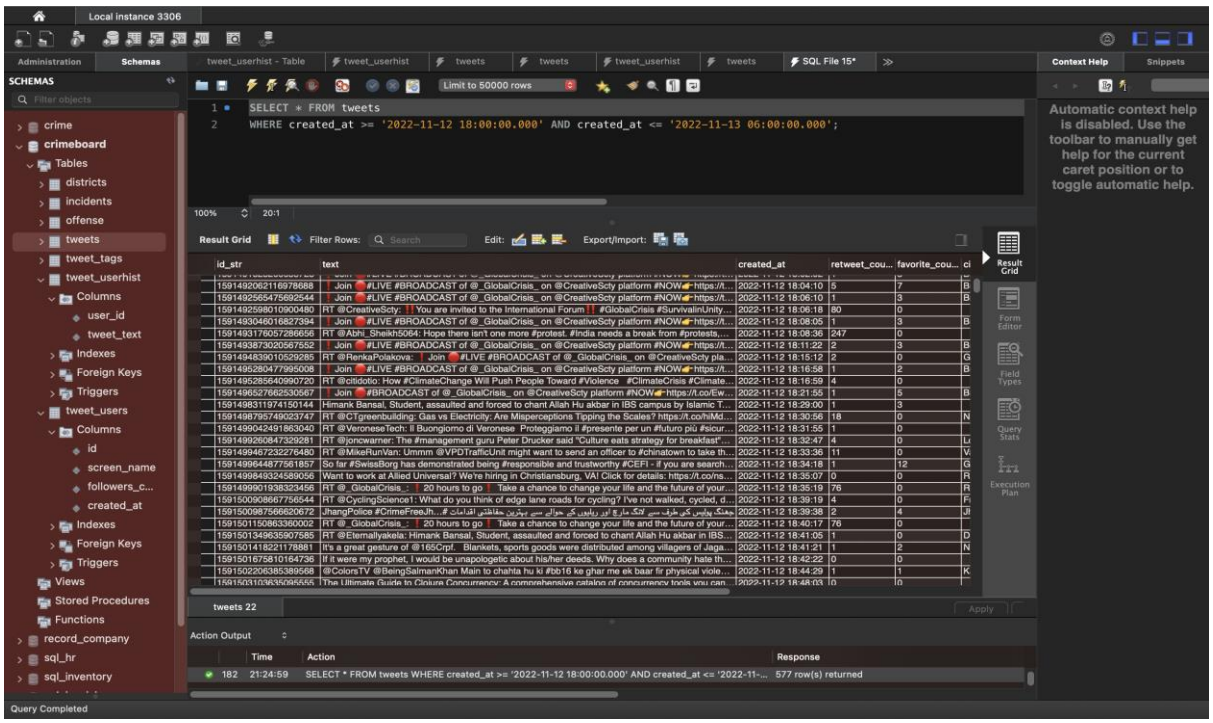
Post Condition: system displays how many tweets about crime were sent between the hours of 6 pm and 6 am.

SQL Query

```
SELECT * FROM tweets
WHERE created_at >= '2022-11-12 18:00:00.000' AND created_at <= '2022-11-13 06:00:00.000';
```

Relational Algebra Expression

σ *created_at >= "2022-11-12 18:00:00.000" AND created_at <= "2022-11-13 06:00:00.000"* *tweets*



17)

Use Case: How many tweets about crime were sent between the hours of 6 am and 6 pm?

Description: Use the tweets table to show how many tweets about crime were sent between the hours of 6 am and 6 pm.

Actor: User

Actor action: The user views how many tweets about crime were sent between the hours of 6 am and 6 pm.

System Responses: how many tweets about crime were sent between the hours 6 am and 6 pm is displayed.

Post Condition: system displays how many tweets about crime were sent between the hours 6 am and 6 pm.

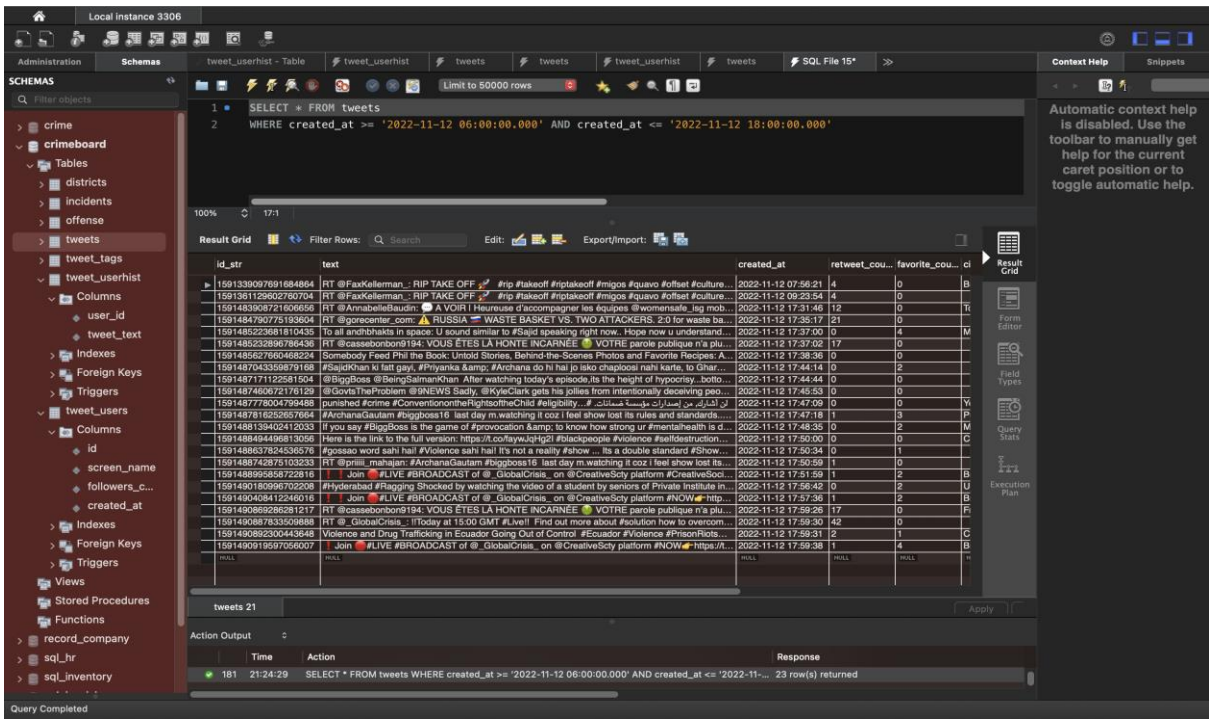
SQL Query

*SELECT * FROM tweets*

WHERE created_at >= '2022-11-12 06:00:00.000' AND created_at <= '2022-11-12 18:00:00.000'

Relational Algebra Expression

σ *created_at >= "2022-11-12 06:00:00.000" AND created_at <= "2022-11-12 18:00:00.000"* *tweets*



18)

Use Case: What tweets has this user posted in the past 24 hours?

Description: Use the tweet_userhist table to show what tweets have this user posted in the past 24 hours

Actor: User

Actor action: The user views what tweets have this user posted in the past 24 hours

System Responses: what tweets have this user posted in the past 24 hours
is displayed.

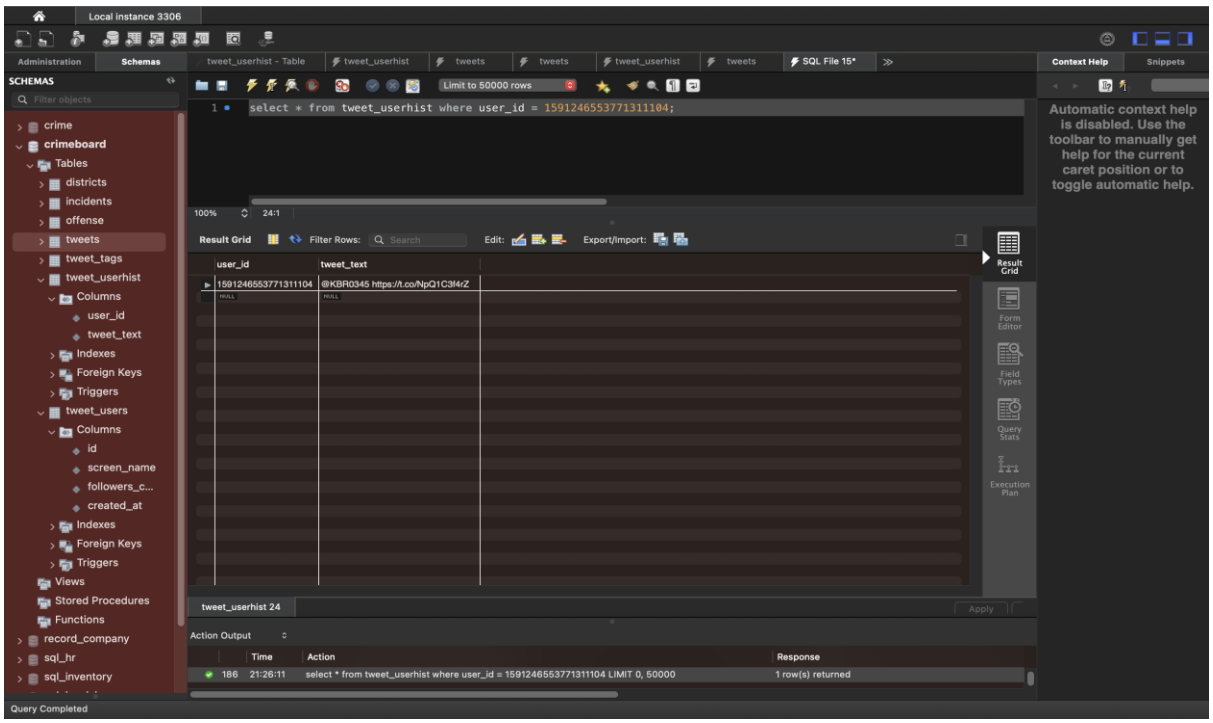
Post Condition: system displays what tweets have this user posted in the past 24 hours

SQL Query

*SELECT * FROM tweet_userhist WHERE user_id = '1572665185319403525';*

Relational Algebra Expression

$\sigma_{user\_id = 1572665185319403525}$ *tweet_userhist*



19)

Use Case: How many tweets has this user posted in the past 24 hours?

Description: Use the tweet_userhist table to show how many tweets have this user posted in the past 24 hours

Actor: User

Actor action: The user views how many tweets have this user posted in the past 24 hours

System Responses: how many tweets has this user posted in the past 24 hours is displayed.

Post Condition: system displays how many tweets have this user posted in the past 24 hours

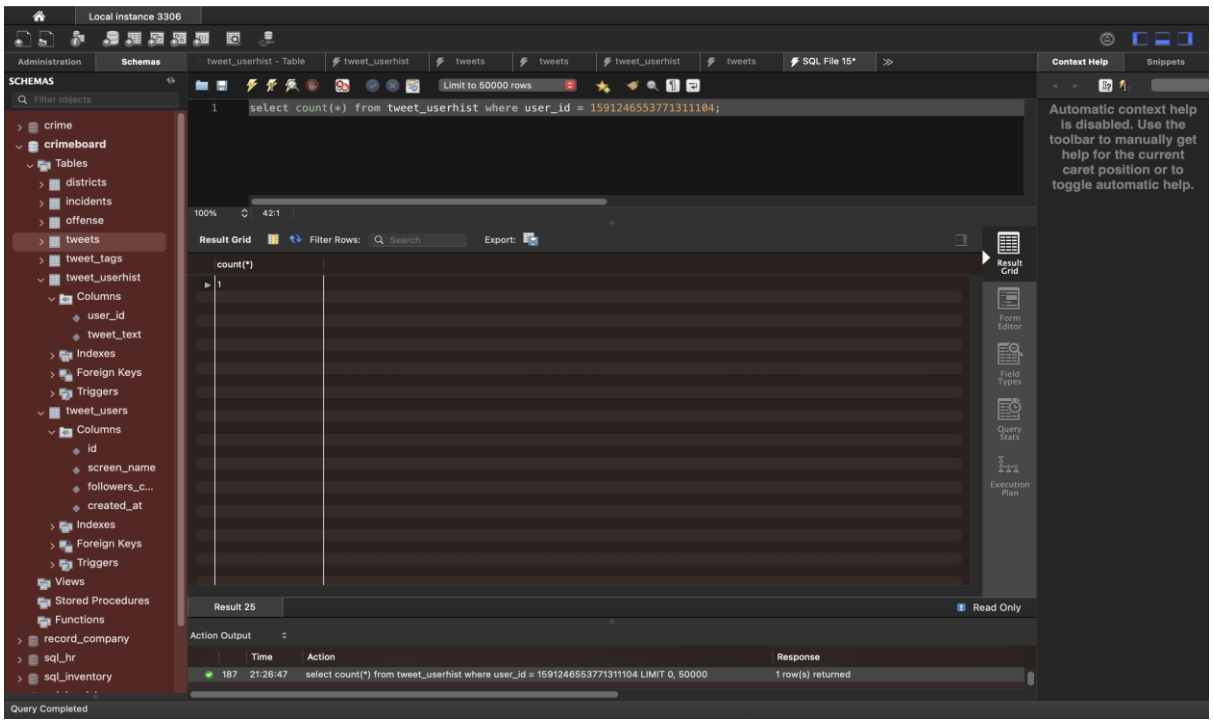
SQL Query

SELECT COUNT(*) FROM tweet_userhist WHERE user_id = 1572665185319403525;

Relational Algebra Expression

π *COUNT (*)*

γ *COUNT (*)*



20)

Use Case: What user posted this tweet about crime?

Description: Use the tweets table to show what user posted this tweet about crime.

Actor: User

Actor action: The user views what the user posted in this tweet about crime.

System Responses: What user posted this tweet about the crime is displayed.

Post Condition: system displays what user posted this tweet about crime.

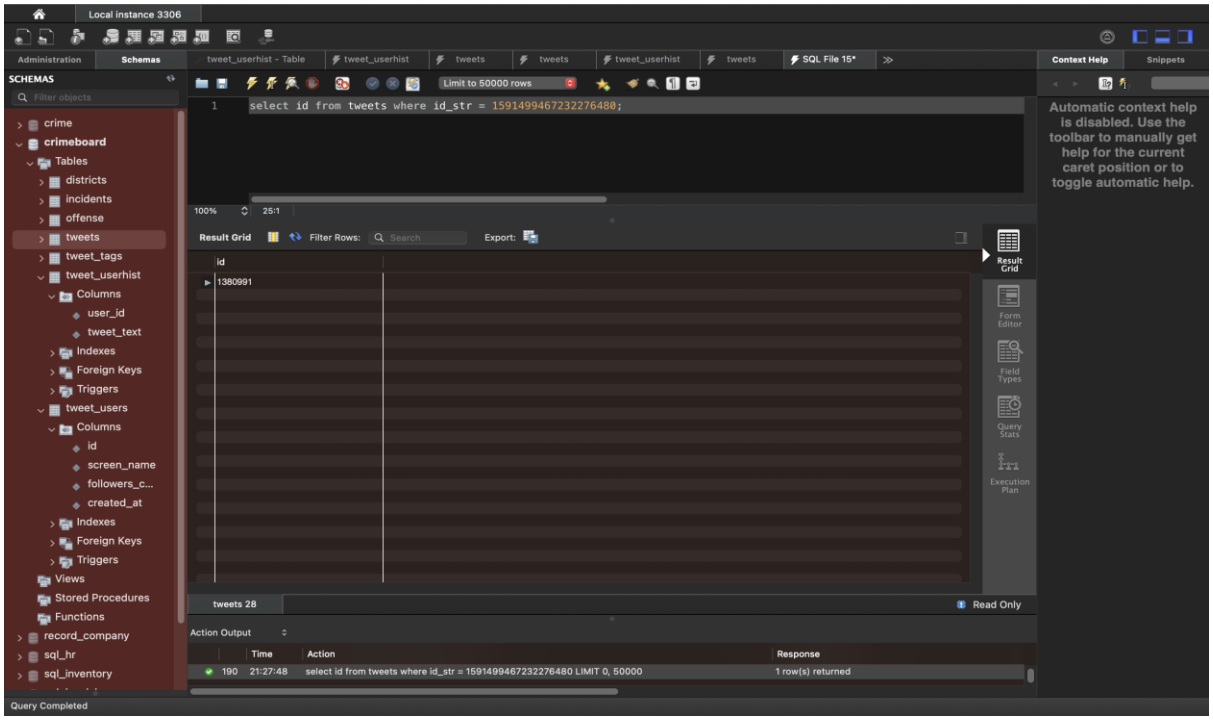
SQL Query

SELECT id FROM tweets WHERE id_str = 1590902258098900992;

Relational Algebra Expression

π *id*

σ *id_str = 1590902258098900992* *tweets*



21)

Use Case: When did the user post this tweet about crime?

Description: Use the tweets table to show when did the user post this tweet about crime.

Actor: User

Actor action: The user views when the user posts this tweet about crime.

System Responses: when did the user post this tweet about the crime is displayed.

Post Condition: system displays when did the user post this tweet about crime.

SQL Query

SELECT created_at FROM tweets WHERE id_str = 1590902258098900992;

Relational Algebra Expression

π *created_at*

σ *id_str = 1590902258098900992* *tweets*